CS770/870 Spring 2017 Animation Basics

Related material

Angel 6e: 1.1.3, 8.6

Thalman, N and D. Thalman, Computer Animation, *Encyclopedia of Computer Science*, CRC Press.
Lasseter, J. Principles of traditional animation applied to 3D computer animation, *Siggraph '87*.
Wikipedia

Preview

- Overview
- Traditional animation
- Keyframe animation
- Procedural animation
- Motion paths
- Articulated animation
- Collision detection
- Dynamics
- Animation languages
- Animation systems

Traditional Animation Overview

- Traditional animation process
  - Animator creates a *storyboard* that outlines the story
    - Just enough sketches to be able to summarize the story
  - Animator draws a series of “key” frames
    - identify all important poses among characters and objects in every scene
  - Junior artists draw all the *in between* key frames
    - called “inbetweening” or “tweening”
  - pre-viewing done with flip charts and incomplete images
  - each *frame* generated and photographed separately
  - first done with pencil drawings
  - when all is ready, final color drawings done

Two ½ D

- Scene background doesn’t change much (or at all) from frame to frame; why re-draw it?
  - Draw background separately
  - Draw foreground on transparent *celluloid* sheet
  - Mount foreground over the background and photograph
  - Called *cel animation* (began in 1915)
- *Multiplane cameras* evolved from *cel animation*
  - Disney built 7-plane camera in 1937
    - planes at varying distances apart, can slide in plane for panning
    - scenes painted in oils on the glass
    - used in *Snow White* and later classics
  - 2 neighboring planes moving in opposite directions can give impression of rotation

Disney 7-plane camera 1937

http://www.michaelspornanimation.com

Principles of Traditional Animation*

- ***Squash and stretch***: shape distortion
- ***Timing***: can affect sense of object and personality
- ***Anticipation***: sets up expectation of coming action
- ***Staging***: presenting an idea so it is unmistakably clear
- ***Follow through* and *overlapping action***: make end of an action very clear as well as its relation to next action.
- ***Straight-ahead action and pose-to-pose action***: choices
- ***Slow in and out***: varying inbetween spacing changes effect
- ***Arcs***: visual path for action needs to be natural
- ***Exaggeration***: helps to clarify action, make it more "real"
- ***Secondary action***: one action leads to another
- ***Appeal***: hardest part

*Lasseter, J., Principles of traditional animation applied to 3D computer animation, Siggraph '87.

Squash and Stretch

- Rigid shape during motion isn't very "realistic"
- Better to exaggerate near discontinuities

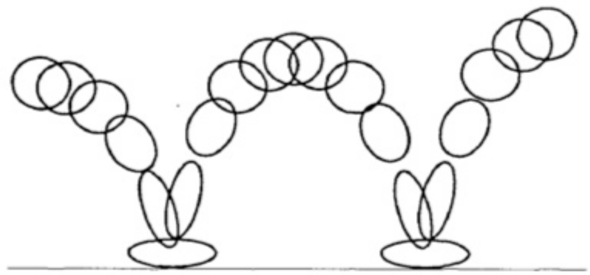


FIGURE 2. Squash & stretch in bouncing ball.

Squash and Stretch 2

- Squash and stretch can also make fast action more realistic and less jerky or strobing
  - Top image: slow movement; overlap makes animation smooth
  - Middle image: very fast motion; can get strobing or jerky motion
  - Bottom image: stretching object so frame positions overlap yields same speed, but smoother

FIGURE 4a. In slow action, an object's position overlaps from frame to frame which gives the action a smooth appearance to the eye.

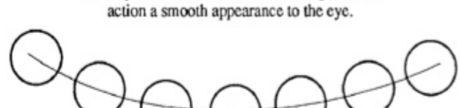


FIGURE 4b. Strobing occurs in a faster action when the object's positions do not overlap and the eye perceives seperate images.

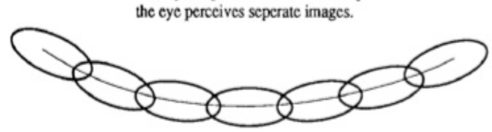


FIGURE 4c. Stretching the object so that it's positions overlap again will relieve the strobing effect.

Timing Matters a Lot

- Speed of action can make huge difference in perceived meaning of the action
  - Audience should
    - anticipate an action
    - understand the action and what it means
    - see and understand the reaction to the action
  - Too much time for any or all of these, audience loses attention
  - Too little time, audience misses the action or its effects
- Timing defines the weight of an object
  - heavy objects accelerate and decelerate more slowly

Timing and Emotion

- Emotional state of a character can be inferred from timing as much as anything.
- Consider two key frames of a head: one looking over left shoulder, the other over the right
  - 0 inbetweens: head hit by tremendous force
  - 1 inbetween: head hit by brick or rolling pin
  - 2 inbetweens: character has a nervous tic or muscle spasm
  - 3 inbetweens: character is dodging a brick
  - 4 inbetweens: character is giving a brisk order: “Move it!”
  - 5 inbetweens: character is more friendly: “Come on over”
  - 6 inbetweens: character sees a beautiful woman or a hunk
  - 7 inbetweens: character tries to get a better look at something
  - 8 inbetweens: character is searching for something
  - 9 inbetweens: character is deep in thought

Actions

- Every action should have 3 parts: anticipation, staging, reaction
- Anticipation
  - include something to be sure audience is looking at the right character for the action that is to occur
- Staging
  - present the action so its goal is completely clear
  - only one idea at a time
- Follow through and overlapping action
  - the action should complete with any appropriate follow through and lead to the next action with reasonable overlap

Slow In and Out

- Uniform spacing isn't realistic
  - vary inbetween timing and
  - use squash and stretch

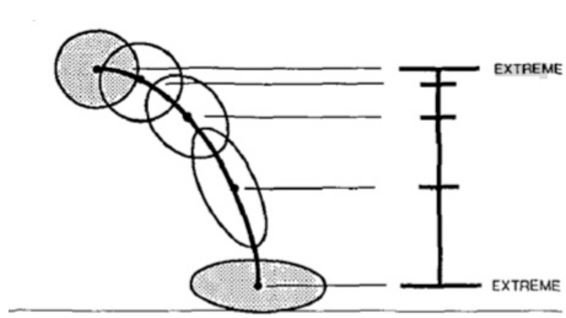


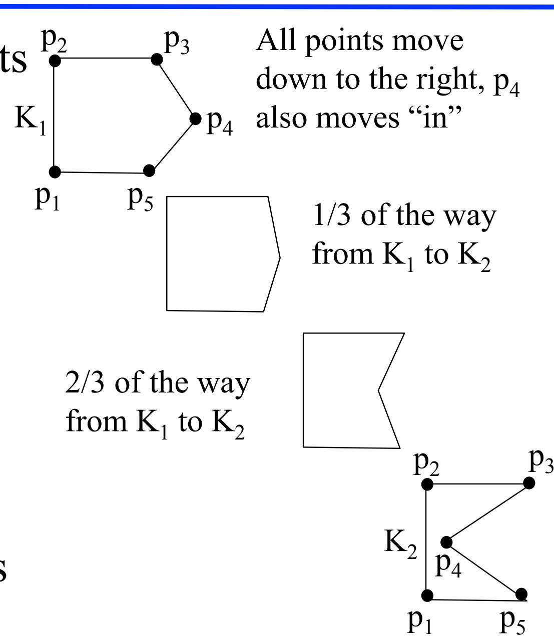
FIGURE 9. Timing chart for ball bounce.

Traditional Computer Animation

- Computer animation software
  - provides tools for interactive specification and previewing of 2D, $2^1/_2$D and 3D animation
  - 3D tools are now very powerful
- Originally were 3 major approaches
  - Image-based key frame animation
  - Parametric key frame animation
  - Procedural animation

Image-based Keyframe Animation

- Identify corresponding points in adjacent key frames
- Linearly interpolate the intervening positions.
- Easy to do, once you have correspondences
- Linear interpolation is not very realistic; cubic splines often used
- Can get more shape changes by introducing new points



Parametric Keyframe Animation

- Can also interpolate parameters of the model rather than just location information
- Parameters could represent almost anything:
  - position: x(t), y(t), z(t)
  - size: radius(t)
  - visualization parameters: color, texture mapping, etc.
- Each key frame needs to identify the parametric value(s) associated with it, so can interpolate between them
- Generally, some form of spline interpolation is preferred, often cubic

Procedural Animation

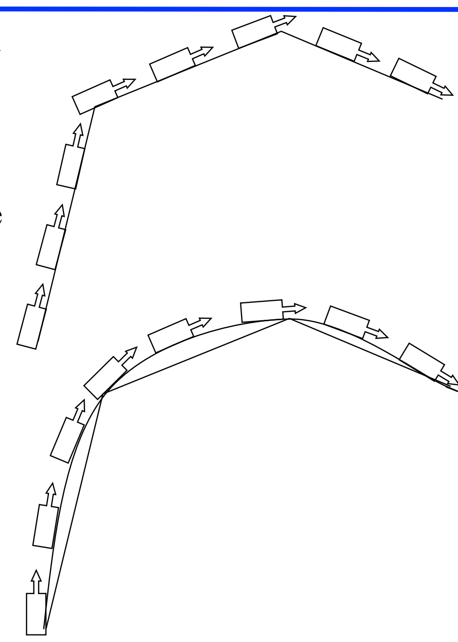
- The motion/shape/characteristics of an object can be controlled by a procedure (or procedures).
- A motion procedure is invoked once per frame and it determines the new parameters for the object for that frame:
  - can use physically correct equations of motion that take weight, gravity, mass, speed, etc. into account
  - can use physically impossible shape transformations (morphing)
  - can use anything you can program!

Motion Paths

- An object may need to follow a specified path in space
  - perhaps, pre-defined
  - perhaps, generated dynamically based on conditions
- Speed
  - could be constant
  - determined procedurally based on the path (is it going uphill or downhill? is it rough terrain or smooth terrain?)
  - based on other factors such as visibility, other objects, etc.
- Is the path on a surface or in space?
- Does the path imply the object's orientation?

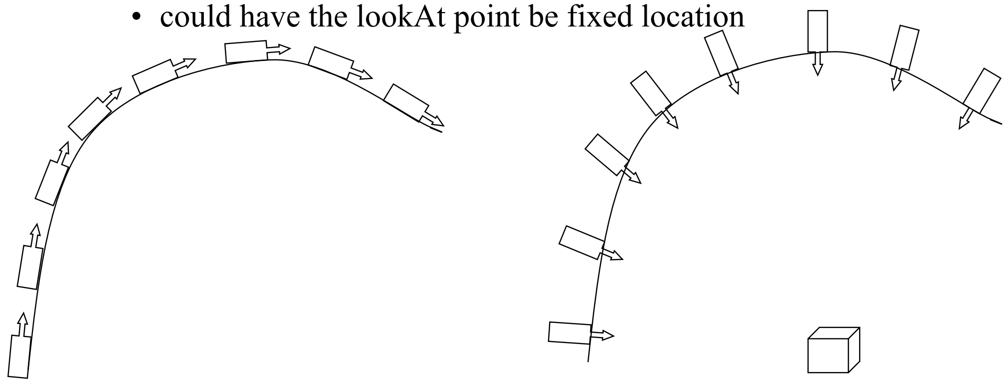
Motion Path Example

- Consider a simple motion path defined by a sequence of positions in space:
  $P_i = (x_i, y_i, z_i)$
- Draw a (spline?) curve through the points
- Move an object's position along the curve at some speed which translates to distance per frame
- Can use *tangent* to the curve to rotate the object so it is heading along the curve
- Relatively easy to create a parametric quadratic or cubic spline through these points
  - and have the object follow the curve



Camera Animation

- Often desirable to move a camera through a scene
  - Specify a motion path for the eye point
  - Specify how other parameters (such as those in *gluLookAt*) should change with respect to the eye
    - could have the eye to lookAt vector be tangent to the curve
    - could have the lookAt point be fixed location



Articulated animation

- Consider a hierarchically defined complex object whose parts can move separately (humanoid)
  - Entire body can be moving
  - Parts can be moving as well
    - some parts may be determining the body motion (legs)
    - some parts may move independent of the body motion, but are still constrained characteristics of the model
- How is it done at the low level?
  - Based on a scene graph hierarchy
  - Some nodes in hierarchy are *joints* with *degrees of freedom* represented by *joint angles*
  - Joint transformations cannot violate constraints on joint angles

Collision detection

- Collision detection is a big deal in real-time
  - brute force is n^2 problem
  - need to use hierarchical bounding box and spatial subdivision techniques
  - special care needs to be taken for small fast objects
    - They could pass over another object in one frame
    - Need a test to make sure that the *path* of an object does not intersect another object in the frame

Kinematic Motion Specification

- Kinematics, or forward kinematics
  - calculate new position of each (part of) an object based on its current position, motion specification and constraints
- Inverse kinematics
  - given current position of articulated body, and desired "final" position (like a key frame) and the constraints, calculate intermediate positions
  - example: animator interactively positions the hand at desired next position, software calculates the required joint angles for the hand, the forearm, the upper arm, the shoulder, etc.
  - requires nonlinear programming

Dynamics

- Kinematics and inverse kinematics do not take physics into account (accept via joint constraints)
  - mass, gravity, inertia, friction, etc.
- More realistic animation is achieved by creating software to simulate these physical laws

Advanced Topics

- Animation languages and systems
- Motion capture animation
- Autonomous behavior
  - avoidance/attraction heuristics
  - swarming
  - other group behavior
- Event-driven procedural animation
- Real-time animation
  - virtual worlds
- Facial animation
- Morphing